Hindawi
Complexity
Volume 2021, Article ID 5161111, 13 pages
https://doi.org/10.1155/2021/5161111

Research Article

Adoption of Convolutional Neural Network Algorithm Combined with Augmented Reality in Building Data Visualization and Intelligent Detection

Minghui Wei,[1] **Jingjing Tang**,[1] **Haotian Tang**,[1] **Rui Zhao**,[1] **Xiaohui Gai**,[1] **and Renying Lin**[2]

[1]*University of Technology Sydney, Sydney 2007, Australia*
[2]*University of Sydney, Sydney 2006, Australia*

Correspondence should be addressed to Minghui Wei; minghui.wei-1@student.uts.edu.au

Received 30 April 2021; Revised 1 June 2021; Accepted 9 June 2021; Published 29 June 2021

Academic Editor: Zhihan Lv

It aims to improve the degree of visualization of building data, ensure the ability of intelligent detection, and effectively solve the problems encountered in building data processing. Convolutional neural network and augmented reality technology are adopted, and a building visualization model based on convolutional neural network and augmented reality is proposed. The performance of the proposed algorithm is further confirmed by performance verification on public datasets. It is found that the building target detection model based on convolutional neural network and augmented reality has obvious advantages in algorithm complexity and recognition accuracy. It is 25 percent more accurate than the latest model. The model can make full use of mobile computing resources, avoid network delay and dependence, and guarantee the real-time requirement of data processing. Moreover, the model can also well realize the augmented reality navigation and interaction effect of buildings in outdoor scenes. To sum up, this study provides a research idea for the identification, data processing, and intelligent detection of urban buildings.

1. Introduction

With the rapid economic and social progression, the number of urban populations is increasing, and various urban management problems have become prominent. Compared with the original urban management methods, the current urban construction needs are far from meeting people's needs [1]. With the support of scientific and technological progression, information technology has led urban management into a new world. As a result, China's urban infrastructure and urban appearance have undergone earth-shaking changes. In particular, the construction of digital cities has become the core of urban progression [2]. The focus of digital city construction is the collection and processing of data in urban facilities, enterprises, shops, and buildings. As the most extensive and common artificial features in urban areas, buildings play an important role in the construction of digital cities [3]. The most common building image detection is based on remote sensing image recognition. From the traditional remote sensing image to the current high-resolution remote sensing image, the geometric structure of the image features is more obvious, the position layout is clearer, and the information of texture and size is more accurate [4]. However, the recognition algorithm for building images is unable to extract effective information from the images due to technical limitations, which leads to incomplete data collected for digital city construction and delays in urban governance and layout decisions [5]. Secondly, as many images cannot effectively correspond to the real city construction, their data processing and detection ability are weak, which further hinders the progression of digital city construction [6]. Therefore, to ensure the degree of building data visualization, improve the ability of intelligent detection, and effectively solve the problems encountered in building data processing, it is necessary to construct and optimize the existing process of building data processing and intelligent detection.

Convolutional neural network (CNN) is a neural network designed and developed based on the neurons of the human body that can effectively extract features from images. This kind of neural network has been applied in many scenarios. The algorithm has the characteristics of fast processing speed and high processing efficiency, so it has been recognized by experts in the field of image recognition [7]. In the building image data processing, due to the lack of effective feature extraction algorithms, the data processing capacity is limited. CNN can just effectively solve this problem, so this method is feasible in building recognition [8]. Secondly, augmented reality (AR) combines computer-generated data such as images, text, and information through specific means to integrate the scene in reality, realizing the interaction between users and data, improving the information carrying capacity of the real world, and enriching the user experience [9]. Augmented reality technology can integrate multimedia or graphic data such as pictures, text, and three-dimensional graphics generated in computers and other equipment into the real-world scene through specific technical means. It allows users to observe and interact with it, thereby further expanding the scope and content of reality that users can perceive, which improves the information carrying capacity of the real world and enriches the user experience. Since mobile augmented reality is not limited to a fixed location when used, it expands the range of users' activities and has high flexibility and scalability. Therefore, applications in outdoor large-scale scenes such as urban environments are becoming more abundant. Typical application scenes include navigation, wayfinding, travel guides, and outdoor entertainment. Mobile augmented reality browsers such as Wikitude developed by Wikitude in the United States and Layar developed by SPRXmobile in the Netherlands are representative products for mobile augmented reality applications in outdoor scenes. Such type of application calculates the attitude information of the mobile phone and combines sensors and satellite positioning to derive the buildings that the user is currently paying attention to. It can also superimpose the corresponding information on the mobile phone screen to achieve the display effect of augmented reality [10]. In summary, mobile augmented reality is widely used in many industries. Its flexibility and universal support for mobile computing devices such as smart phones enable it to allow more users to participate. Due to the broad application prospects of mobile augmented reality, the research on mobile augmented reality methods, especially mobile augmented reality for outdoor large-scale scenes, has very practical significance.

In this study, large-scale high-resolution remote sensing images freely available on Google Earth are used. Deep learning technology is used to identify the building area in the image, and image processing technology is used to extract the building outline from the identified building area. Then, a set of lightweight target detection model SqueezeNet SSD is designed for mobile. The model is based on the deep learning model SSD, combined with the SqueezeNet lightweight convolutional neural classification network to reduce network complexity and increase operating speed. Moreover, deep transfer learning strategy is adopted to train the model to realize intelligent general detection of buildings. The main innovations are as follows. First, effective integration of CNN algorithm and AR is implemented to ensure the accuracy and interactivity of model recognition. Second, the proposed model avoids the need for traditional algorithms to establish image feature matching libraries and the limitations of only identifying specific buildings and improves versatility. Third, the lightweight of the model is realized so that it can run on the mobile terminal, reduce network dependence, and ensure flexibility and real time.

This work includes five parts. The first part is the introduction, which puts forward the importance of research on building identification and detection and determines the main research ideas. The second part is a literature review. Through the analysis of the research status of CNN in the field of building recognition and the research status of AR in the field of building recognition, the existing problems in the current research are clarified, and the appropriate research ideas are determined. The third part introduces the research methods, proposes a building target detection model based on CNN and AR, and explains the details, parameters, and datasets of specific modeling. The fourth part discusses the research results, analyzes the examples of the proposed model, draws out the performance and advantages of the model, and compares the proposed model with other algorithms. The fifth part gives conclusions, including actual contributions, limitations, and future prospects.

2. Related Work

2.1. CNN in Building Recognition. Many scholars have reported on the adoption of CNN in building recognition. Xiao et al. proposed a globally supervised low-rank expansion method and a CNN model with adaptive weight reduction technology to solve the problems of low speed and small storage in building recognition. It was found that the proposed algorithm can surpass the current best neural network. Compared with the latest CNN model, this method was about 30 times faster and the cost-effectiveness increased by 10 times [11]. Yan et al. proposed the graph convolution neural network (GCNN) architecture to analyze the spatial vector data of the graph structure and found that GCNN produced satisfactory results in terms of identifying regular and irregular patterns. Compared with the method, there was a significant improvement [12]. Wang et al. proposed an effective method based on deep CNN to solve the problem of low efficiency and poor accuracy of traditional algorithm recognition and adopted a new DenseNet for building recognition. It was found that the new network not only maintained the main performance advantages of DenseNet, but also effectively reduced memory consumption. In addition, the algorithm model improved the background noise and character adhesion to 99.9% [13]. Wei et al. proposed a multistream CNN framework to improve the accuracy of recognition by learning the correlation between a single building and the map. It was revealed that the proposed multistream CNN framework was superior to the latest method based on sEMG [14]. Yao et al. found that the learning ability of the model was greatly improved by

using CNN combined with a deep belief network. Regardless of whether it was a dynamic image or a static image, the accuracy of the model was improved [15].

2.2. AR in Building Detection. There are few researches on the adoption of AR in building inspection, mainly because this method requires a high cost, and there is still no well-established method for it. Zhou et al. used AR in the construction of building tunnels, by which on-site quality inspectors can retrieve virtual quality control benchmark models that can be established according to quality standards and can automatically evaluate structural safety by measuring the difference between the baseline model and the actual facility view [16]. Mylonas et al. built a university Internet of Things through reality augmentation algorithms. The model can effectively improve the teaching efficiency of teachers and the teaching effect of students. In addition, the method was applied to other teaching and got favorable results [17]. Chen et al. combined building information model and AR to facilitate the inspection and maintenance of building features through this model, thereby overcoming the limitations of paper documents on these tasks [18]. The combination of information and real objects through AR effectively promotes the presentation of information in an instant, visual, and convenient way. García-Pereira et al. proposed an AR tool designed to assist inspectors, perform collaborative inspections, and obtain annotations of multiple types and geographic locations. The final result also confirmed that the model can well perform assessment of the quantitative and qualitative performance of the building in the real environment [19]. Liu et al. fully integrated ARAR in building information modeling (BIM) and proposed a BIM+AR construction method. It was found that this model can effectively solve the problems of assembly errors and low communication efficiency [20].

2.3. Summary of Research Issues. In view of the above research, it is found that many scholars have made effective improvements to its algorithm performance in the existing CNN building recognition, and they have also obtained good experimental results. However, the actual adoption system for unconstrained real urban scenes does not yet exist; in particular when dealing with building areas with a large dynamic range, intricate textures, and large occlusion areas, the extraction accuracy is significantly reduced. Therefore, freely provided large-scale high-resolution remote sensing images are utilized, CNN is applied to identify the building area in the image, and image processing technology is employed to extract the building outline from the identified building area. In AR building detection, since the satellite positioning accuracy is easily affected by external conditions, inertial sensors usually have certain errors, which limit their scope and flexibility and have a certain impact on the user experience. Therefore, AR is optimized to a certain extent, and the visualization and intelligent detection capabilities of building data are enhanced through the fusion of the two algorithms.

3. Research Methodology

3.1. Building Data Visualization Model. Based on the above research issues, the following system is designed, as presented in Figure 1. The prototype system is composed of four parts: a vision detection and tracking module, a posture positioning and geometric calculation module, a building information database, and an AR display and interaction module. Visual inspection and tracking module is responsible for processing the image data passed by the camera and detecting buildings from it. Then, the detected buildings are followed up for rapid parallel tracking. CNN model is utilized to detect buildings in the current picture, and a detection-tracking error recovery mechanism is established. Posture positioning and geometric calculation module is mainly responsible for providing users with the positioning coordinates of the current position and the posture data of the mobile phone sensor. It also calculates geometric information such as distance and azimuth based on the attitude and positioning data. Building information database provides relevant information corresponding to the building, including building data and customizing building information data. In the building information, the latitude and longitude coordinates can participate in the calculation of distance and azimuth. Other types of attribute information are used to generate virtual objects to provide support for the virtual and real fusion. AR display and interaction module is mainly used to construct the camera coordinate system and process the coordinate conversion between the screen coordinate system and the camera coordinate system. In addition, the building information is made into virtual objects in real time and rendered in the designated position in the camera coordinate system frustum, to realize the real-time interaction between the user and the system.

First, the system preprocesses each frame of image data captured by the camera through the visual inspection and tracking module, which is converted into an image of the specified size and format and input into the trained SqueezeNet single shot MultiBox detector (SSD) target detection model (Figure 2). Then, for each item in the target detection result, a target tracker is created, and the same frame of image and the target's detection frame result are used to perform the initialization operation, so that the parallel tracking of all target detection results is realized. Then, the user's current latitude and longitude coordinates and mobile phone posture data are determined with the aid of posture positioning and geometric calculation modules. Through the target matching method based on the azimuth relationship, the screening of building targets and the precise matching of target detection results and building information are realized combined with the building longitude and latitude coordinate information in the building information database. The virtual objects are made based on the data in the building information database. AR display and interaction module are adopted to instantly render the virtual objects, realize hybrid virtual and real registration and provide interactive support, and provide users with AR navigation and interaction for buildings.

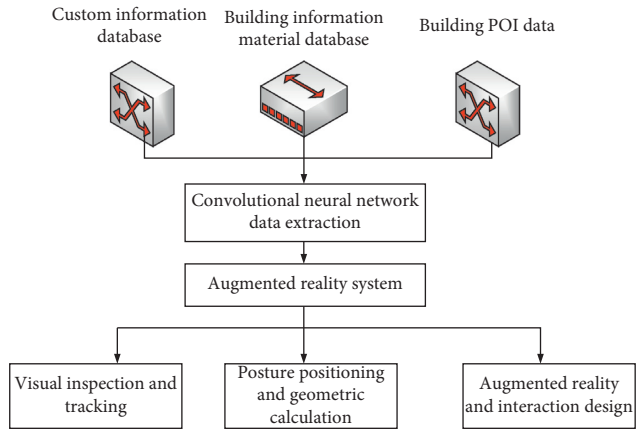

Figure 1: Building data visualization model.

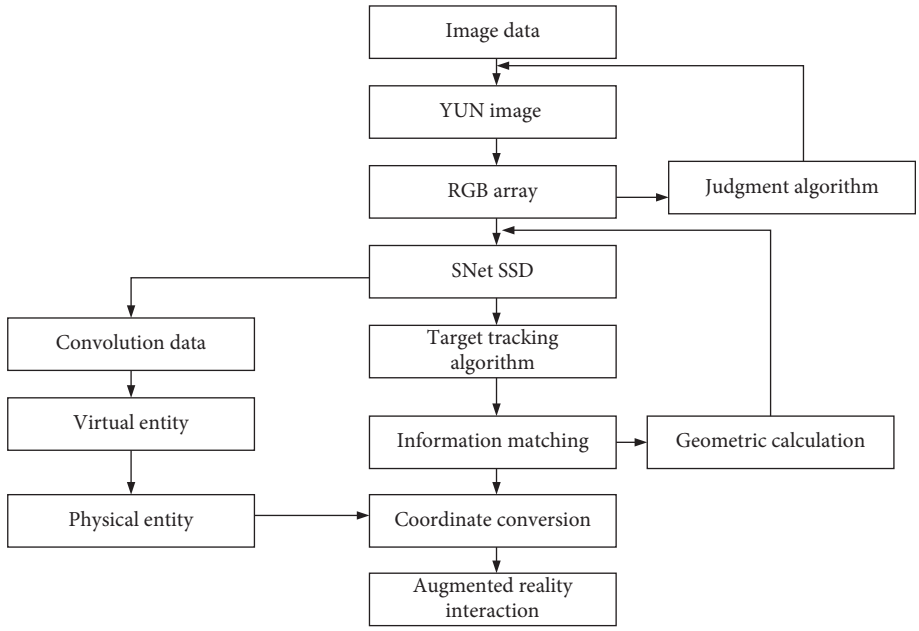

Figure 2: AR prototype system operation process.

The SSD target real-time detection model is a multiscale prediction (Multiscale) network, as shown in Figure 3 below. The model predicts from different feature maps and then merges the prediction results together. SSD is compared with YOLO, and it is found that YOLO is of a single scale. In terms of frame generation, SSD is also in place in one step, and the frame is generated and tested at the same time. Therefore, real-time detection of pictures is realized. There are a total of six scales, and frame generation and prediction are performed on each scale. Finally, nonmaximum suppression (NMS) is used to filter and get the result. The main difference from YOLO lies in the multiscale, backbone part, full convolution, and that the prediction part is also convolution (YOLO's prediction part is a fully connected layer). Compared with YOLO, SSD uses CNN to directly perform detection instead of performing detection after the fully connected layer like YOLO does. Using convolution to detect directly is just one of the differences between SSD and YOLO, and there are two other important changes. One is that SSD extracts feature maps of different scales for detection. Large-scale feature maps can be used to detect small objects, and small-scale feature maps can be used to detect large objects. The second is that SSD uses a priori boxes (prior boxes, default boxes, called anchors in Faster R-CNN) of different scales and aspect ratios. The disadvantage of the YOLO algorithm is that it is difficult for it to detect small targets and the positioning is not accurate, but these important improvements enable SSD to overcome these shortcomings to a certain extent. SSD takes VGG16 as the basic model and then adds a new convolutional layer based on VGG16 to obtain more feature maps for detection.

3.2. CNN for Data Feature Extraction. Since the SSD model has the characteristics of fast running speed and high detection accuracy, it is selected as the algorithm used in

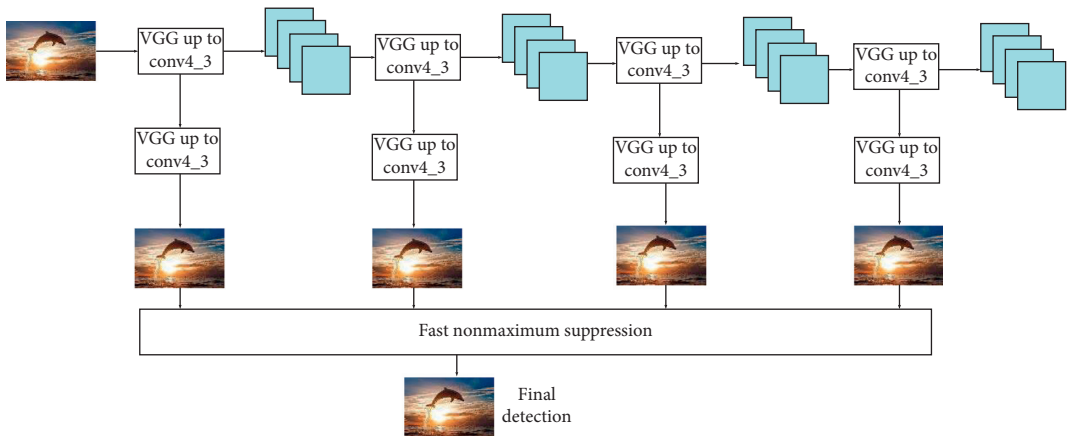

Figure 3: SSD target real-time detection model.

building target detection. To solve the problem of excessive model calculation and excessive weight volume, the model is deployed on the server side, and the real-time operation of the model algorithm is realized through the high-performance processor on the server side. The image obtained by the mobile device is compressed and uploaded to the server to be handed over to the model for target detection through an established instant connection between the server and the mobile. Then, the target detection results obtained are transmitted back to the mobile terminal device [21]. First, the VGG-16 basic classification network with the largest amount of calculation in the SSD model is replaced with the SqueezeNet network with the classification layer removed. SqueezeNet is a relatively small deep CNN, which reduces the number of parameters of the network to one-fifth of that of AlexNet while ensuring that it has the classification accuracy of the AlexNet network. The network structure of SqueezeNet convolutional neural classification network is shown in Figure 4. The network first uses a common convolutional layer to perform convolution operations on the input image, Input, to extract features. Then, it employs the ordinary convolutional layer to calculate again and performs global average pooling on the final feature map.

The structure of the SqueezeNet SSD target detection model combined with the SqueezeNet structure is shown in Figure 5. This structure uses the SqueezeNet network with the classification layer removed instead of the VGG-16 network as the basic classification network of the model, and the subsequent additional layers are modified accordingly to match it. The number of convolution filters in the inner convolution layer is halved, and the global average pooling layer is connected after the Ex3_2 layer.

3.3. AR Intelligent Detection. The camera coordinate system space is established based on OpenGL ES 2.0, and the scene renderer is constructed. The frustum and visible area are set, and the production and registration of virtual objects are implemented, so as to realize the virtual and real fusion display of AR. OpenGL ES is a powerful and convenient low-level 3D graphics library designed for mobile devices. The first step of AR display is converting the building detection frame on the screen coordinate system to the corresponding position of the target plane of the frustum of the camera coordinate system. The calculation of the distance between the target plane and the camera involves the calculation of the latitude and longitude distance and the azimuth angle. The longitude and latitude coordinates of the building are acquired from the building information database, and the posture estimation and positioning module are used to obtain the positioning coordinates and posture data of the device. Then, it combines the *AMapUtils.calculateLineDistance* method of AutoNavi SDK to calculate the geometric distance [22]. After the coordinate conversion is finished, virtual objects need to be generated based on the data in the building information database. Each building is approximated by a cylinder object with a height and diameter similar to it under the current viewing angle, along with several floating information window *infowinObject* and a floating coordinate window *coordObject*. These classes all implement a method for resetting parameters and position information, for modifying the vertex coordinates of their own graphics according to the latest building detection frame coordinates, etc., and implement a draw method for drawing. The GL10 object with relevant drawing parameters is configured for drawing. Finally, in the *onDrawFrame* method of the renderer class inherited from *GLSurfaceView.Renderer*, all virtual objects are traversed and drawn one by one at the designated position in the frustum, so as to be superimposed and displayed on the real picture of the mobile phone camera to realize virtual and real fusion [23].

3.4. Data Source and Performance Analysis

3.4.1. Development and Hardware Environment. The prototype system is developed for the Android platform, and the Ubuntu Desktop 16.04 LTS 64 bit system is used to implement the development process. The development tool used is Android Studio 3.2, and the development language is Java. Given the system compatibility, the Android SDK version used is LOLLIPOP (Android 5.0, API 21). The database uses SQLite 3.7.11, and the graphics library uses

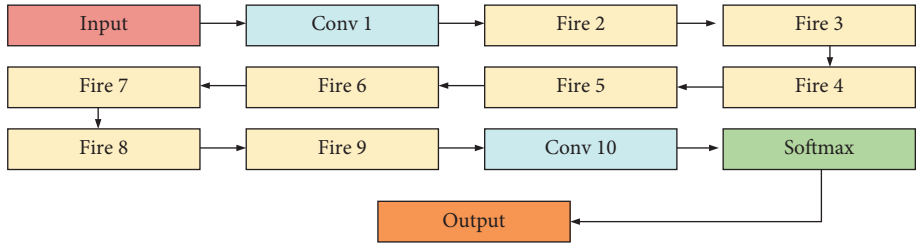

Figure 4: SqueezeNet CNN structure.

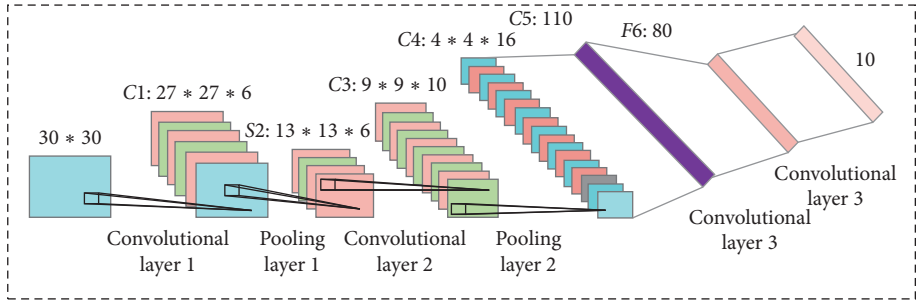

Figure 5: SqueezeNet SSD object detection model structure.

Open GL ES v2.0. The computer vision algorithm library uses OpenCV 3.1.0, and the algorithm support in OpenCV_Contrib is imported. The SqueezeNet SSD target detection model is built based on the MXNet deep learning framework. CUDA 9.0 is taken as the GPU parallel computing architecture, and CUDNN 7.0 is used to achieve GPU-based neural network computing acceleration. The weights of the basic classification network of SqueezeNet use ImageNet-based pretraining weights. The model is written in Python language and completed deep transfer learning training on a computer equipped with GPU. The CPU of the computer used for system development and model training is Intel® Core™ i7-6700K CPU @ 4.00 GHz × 8 (8 GB memory), and the GPU is NVIDIA GeForce GTX 1060 (6 GB video memory). The CPU configuration of the mobile phone equipped with the prototype system is Qualcomm Snapdragon 821 (6 GB RAM). The camera uses Bublcam, which is a relatively small panoramic camera. With a software package, Bublcam is designed for mainstream consumers who want to buy an easy-to-operate virtual reality video device.

3.4.2. Dataset. In the experiment, the Massachusetts remote sensing dataset provided by Mnih is used, abbreviated as Mass.Buildings. The dataset contains 151 remote sensing images of the Boston area. Each image has a resolution of 1500 × 1500 and corresponds to 2.25 square kilometers. The entire dataset covers about 340 square kilometers. The entire dataset is split into a training set containing 137 images, a test set of 10 images, and a verification set of 4 images. Since the training image is very large, directly using it for training will cause memory overflow. To train this network, a sliding window with a size of 256 * 256 and a step size of 64 is used to cut out a series of image blocks from each visible light remote sensing image. Some areas of the image in this dataset are empty, so it is necessary to determine the ratio of pixels with pixel values (255, 255, 255) in the cropped image block to this image pixel when sliding window cropping is used. If this ratio exceeds 0.02, the image block will be discarded. The labeled binary image block at the same position of the artificially labeled image is cut out in the operation of remote sensing images. After cropping, a training set with 75,938 image blocks and a validation set with 2500 image blocks in total are obtained. To compare with previous methods, 10 images with a resolution of 1500 × 1500 are used as the test set. What needs to be explained here is that the cropped visible light remote sensing image block does not make any linear changes in the pixel value; that is, it is still expressed in RGB format. The gray value range of each channel is 0–255. The gray value range is changed to 0-1 for artificially labeled images; that is, the pixels belonging to the building are marked as I, and the pixels not belonging to the thousand are marked as 0. This marking method is the most commonly used method in the field of semantic segmentation. If the label requires to be visualized, the reader only needs to multiply the image by 255 to solve the classification problem of thousands of N categories. Each tag value can be read in turn, and N single-channel binarized images can be used for visualization. To speed up the data reading speed, the training set and the verification set composed of the cropped image blocks are written into the LMDB library through the data stream. Two points need to be paid attention to when the database is constructed. First, the number of a pair of corresponding remote sensing images and artificial label images should be the same. Second, a randomization function should be used to scramble the image blocks that are cropped in order.

3.4.3. Performance Verification. To verify the actual performance of the lightweight target detection model proposed, a common method of evaluating the target detection model is adopted. The training set of PASCAL VOC2007 and VOC2012 target detection dataset [24] are utilized to train the model. The detection accuracy of the model is tested through the test set of the PASCAL 2007 dataset, which is compared with the indicators of other models to evaluate the performance of the model. The model performance is determined regarding the accuracy and recall rate, and the average accuracy rate is used to analyze the target detection effect to verify the generalization ability of the model. The trained model needs to be run in the PASCAL VOC2007 test dataset, and the specific calculations are as follows:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{1}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{2}$$

In equations (1) and (2), FP stands for false positive samples, FN stands for false negative samples, and TP stands for real samples. The samples are collected by judging the intersection over union (IOU) between the detection frame generated by the target detection model and the real frame. The IOU threshold is given, and all detection frames whose IOU with the real frame are greater than 0.5 are adopted as samples [25]. The calculation of IOU is as follows:

$$\text{IOU} = \frac{\text{Box}_{\text{detection}} \cap \text{Box}_{\text{Ground truth}}}{\text{Box}_{\text{detection}} \cup \text{Box}_{\text{Ground truth}}}. \tag{3}$$

In equation (3), $\text{Box}_{\text{detection}}$ is the detection frame range, and $\text{Box}_{\text{Ground truth}}$ is the real frame range. For this curve, the highest accuracy value is selected in each recall interval to get a series of accuracy values. The average precision (AP) is obtained by averaging these accuracy values. The specific calculation is as follows:

$$\text{AP} = \int_0^1 p(r)\text{d}r. \tag{4}$$

In equation (4), $p(r)$ is the average of precision. The image recognition detection speed is mainly used to reflect the performance of the system detection, and its calculation equation is as follows:

$$W = \frac{D}{T}. \tag{5}$$

In equation (5), D is the pixel distance, T is the system running time, and W is the system detection speed.

4. Result Analysis

4.1. Algorithm Performance Comparison. Figures 6(a) and 6(b) illustrate the loss curve and the detection accuracy of the model proposed after training on different datasets, respectively. As the number of training rounds increases, the accuracy of the training phase and the accuracy of the verification phase of the model continue to improve, while the loss value of the training phase and the loss value of the verification phase of the model continue to decrease. The accuracy and loss values basically converge after 800 rounds. After the training phase is over, the weight parameter file volume of the obtained model is significantly smaller than the weight parameter file volume of the original model. Its compact structure and volume occupancy make it more suitable for embedded or mobile devices.

Figure 7(a) is the AP of the model in the target detection category from Aeroplane to Dining table, and Figure 7(b) shows the AP of the model in the Dog to TV monitor target detection category. The various AP values calculated by the model on the PASCAL VOC 2007 target detection test set are basically stable, and the model's mAP value is 53.7%, which means that the model has high detection accuracy for different target detection categories.

The latest research algorithm [26–28] is compared with the algorithm proposed in Figure 8. Figure 7(a) shows the accuracy of different models on different datasets, and Figure 7(b) shows the recall of different models on different datasets. Figure 7(c) shows the generated file sizes of different models, and Figure 7(d) shows the number of frames per second processed by different models on different processors. Compared with other models, the proposed SqueezeNet SSD model has obvious advantages in accuracy. It is because the lightweight structure takes a smaller size image as input, and the performance of the basic classification network used is weak. Moreover, the number of multiscale feature maps selected is less than that of the original model. However, the recall rate has obvious advantages when the dataset is 250. Compared with the model FPGA in latest research, the performance of the proposed one is increased by 25%. The proposed model generates the smallest proportion in the size of the file generation, only 17.8 Mb, which is about 5 times the speed of the original SSD model. Under different GPU and CPU conditions, the algorithm model shows a high processing speed.

In Figure 9, to verify the performance of the HF-FCN algorithm, it is compared with two methods that use the same dataset and use deep learning. The performance of the different methods is very similar, but they are far lower than our method. Under the same accuracy (0.7), the proposed method always has a higher recall rate of 95%. To make the measurement standard more stringent, the relaxation threshold p is set to 0. It is found that the proposed model also has the same trend. This method is obviously better than these four methods. Based on this, it is concluded that the method proposed in this study is relatively more suitable for recognizing building areas in complex scenes.

4.2. Building Outline Estimation. In Figure 10, ten different building types are estimated, where Figure 10(a) is the time-consuming result of a small-sized building, and Figure 10(b) is the time-consuming result of a large-sized building. When there are many large-size buildings, the time consumption increases greatly, reaching 10 seconds. For remote sensing images with a large proportion of small buildings, the estimation time of the outline of small buildings increases

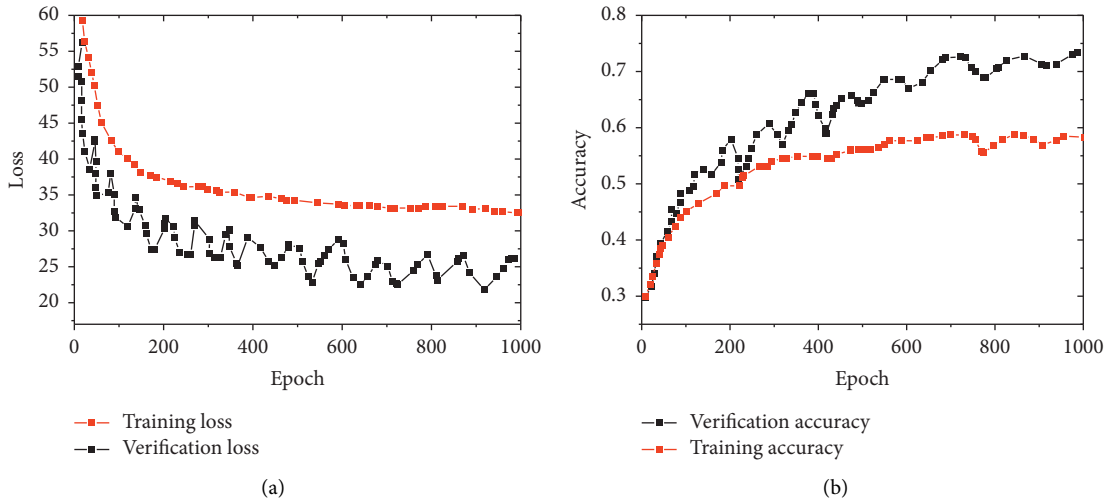

Figure 6: The loss and accuracy of the model after training on different datasets.

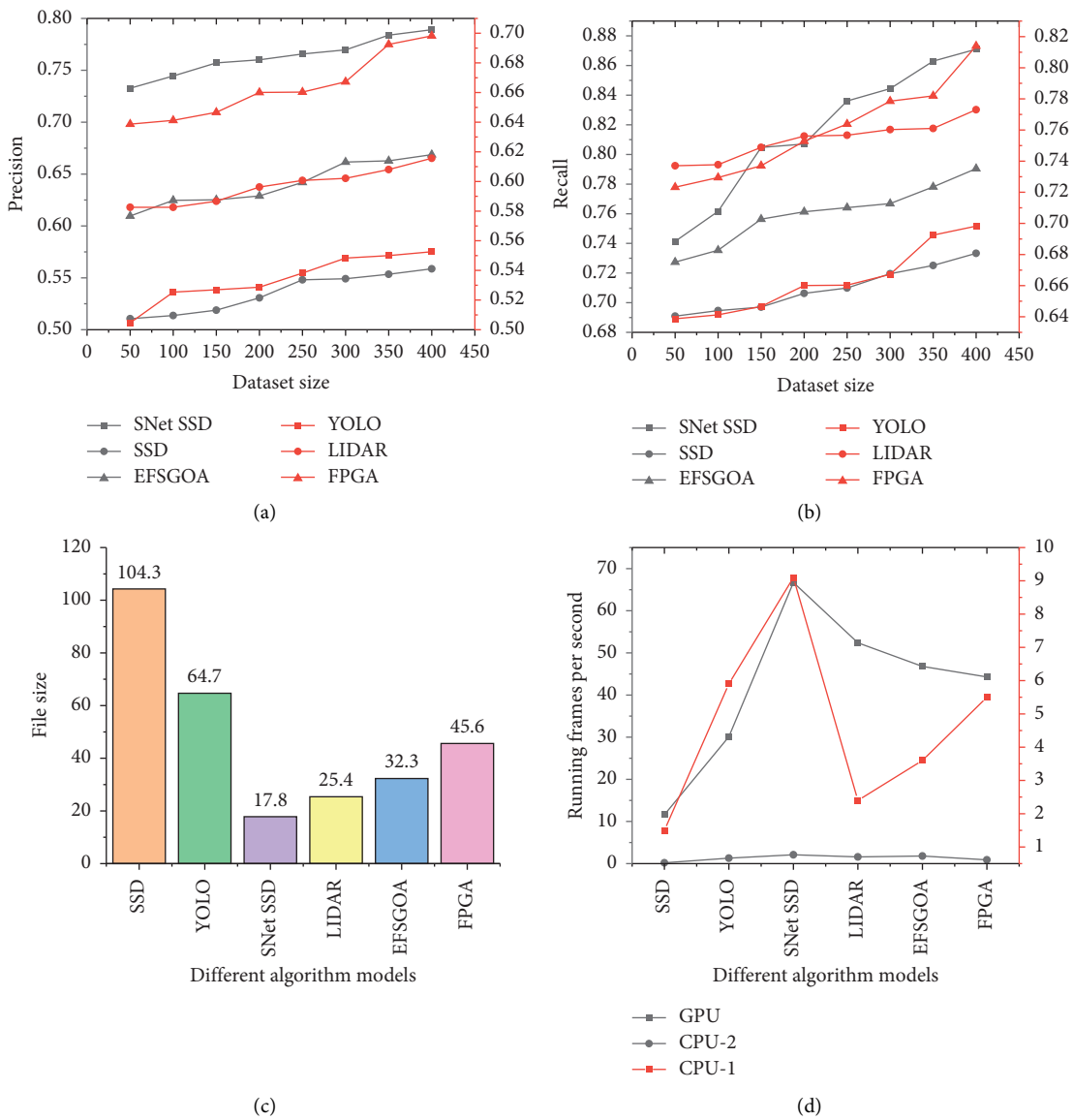

Figure 7: Comparison and analysis of the performance of different models.

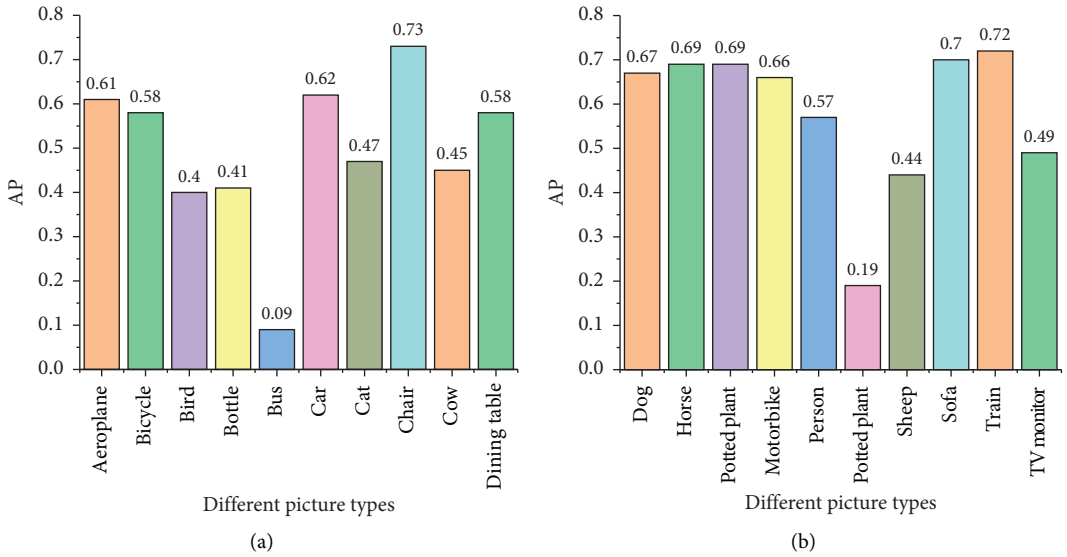

Figure 8: Target category and corresponding AP in model test results.

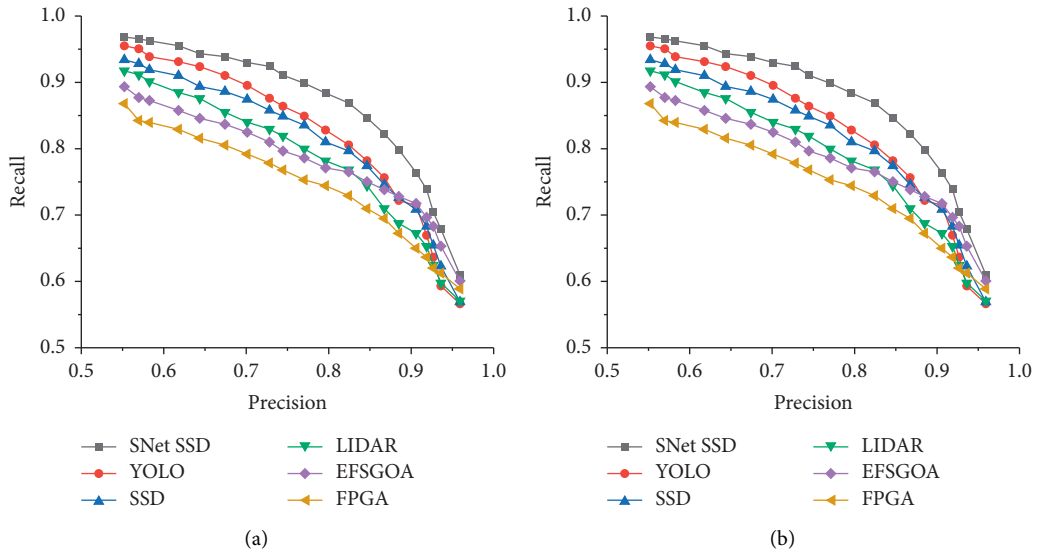

Figure 9: Performance analysis of different models under different parameters.

dramatically, and the total time reaches 7.6 seconds. For image has fewer small-sized buildings and few large-sized buildings, the overall time is only about 4 seconds.

4.3. Building Data Visualization. Figure 11 is a screenshot of the results of the detection and tracking of some buildings. SqueezeNet SSD model running on the mobile terminal can accurately detect various types of buildings and has a high confidence. When multiple buildings appear on the screen, the model can also detect them separately and performs parallel tracking.

Figure 12 is the placement and virtual-real registration effect of the cylinderObject in the system in the frustum. Aided by the relevant coordinate conversion of the detection frame coordinates, the height and width of the cylindrical object in the camera coordinate system are close to the real building target. Its approximation is utilized to replace the building target model, and relatively accurate virtual and real registration with the real scene is realized.

4.4. Evaluation of Detection Results. In Figure 13, the time-consuming process of different detection procedures is analyzed, and it is found that the target detection task takes the longest time due to its relatively large amount of calculation. However, since the system uses the target tracking algorithm, the target detection task has little effect on real-

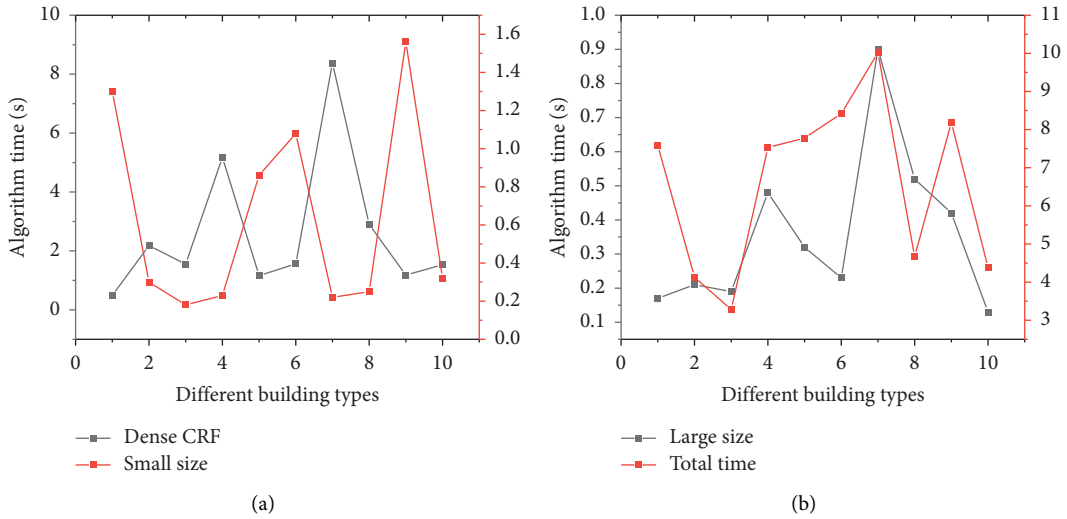

Figure 10: Building outline estimation results.

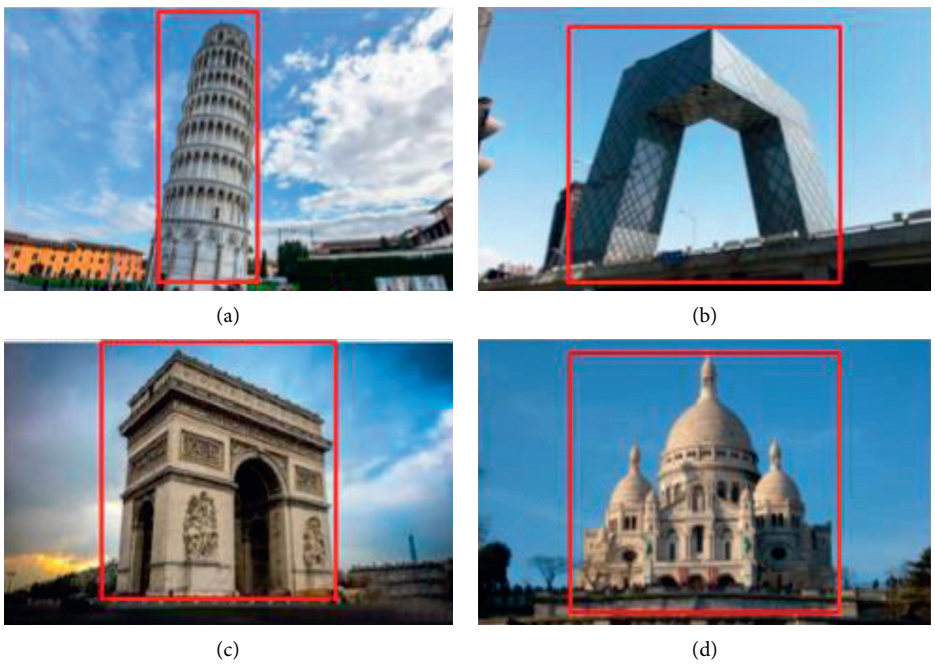

Figure 11: Screenshots of the results of detection and tracking of some buildings.

time performance. The operating speed of the system reaches 11 FPS. The system proposed has good real-time performance compared with the traditional image recognition and detection speed of 500 FPS.

Figure 14 is the statistical results of the detection and matching accuracy of some buildings. Figure 14(a) is the main high-rise buildings, and Figure 14(b) is the main residential buildings. It is found that the detection confidence of most buildings is basically maintained above 0.80. In addition, the confidence level of some buildings even reaches about 0.95, proving that the target detection model can achieve better detection results for most buildings. In general, the system can accurately detect buildings and can perform building information matching, with an average detection confidence of 0.896 and an average matching accuracy of 96.92%, which basically meets the requirements.

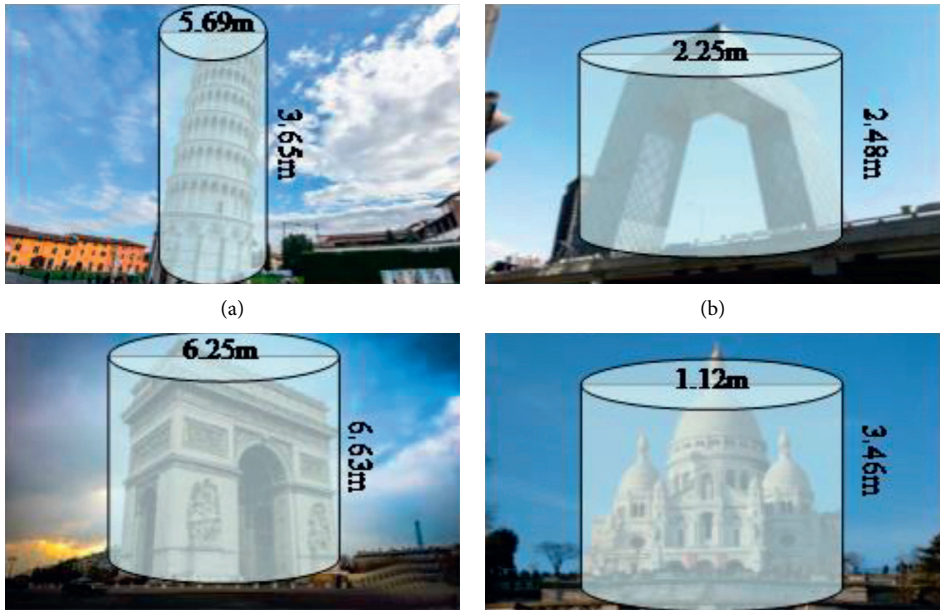

Figure 12: Placement of the cylindrical object in the frustum and the virtual-real registration effect.

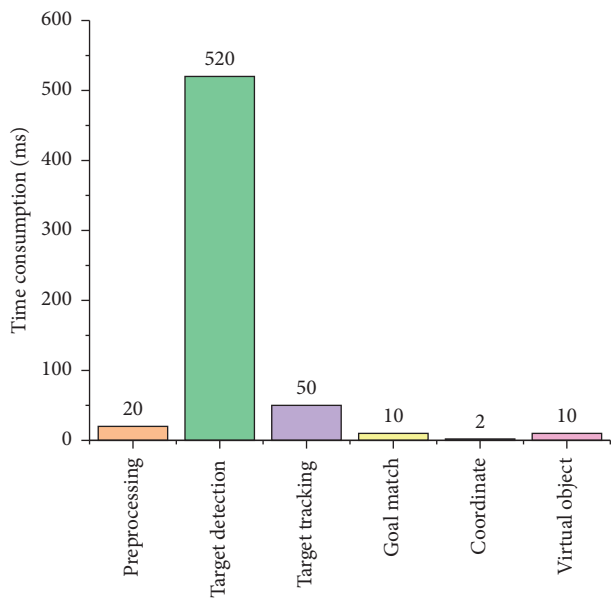

Figure 13: Statistical results of average time-consuming system tasks.

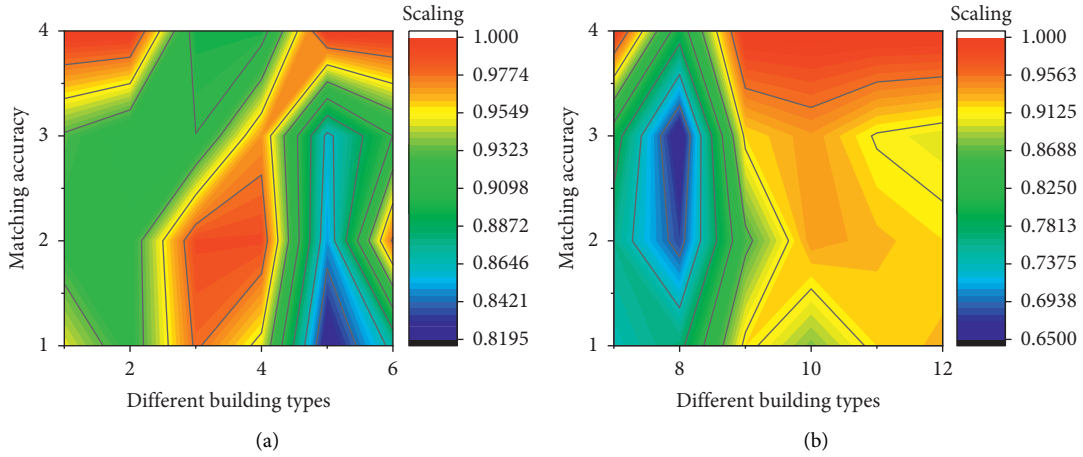

Figure 14: Statistical results of detection and matching accuracy of some buildings.

5. Conclusions

In this study, a mobile-oriented intelligent building detection model is designed first, and a mobile augmented reality hybrid registration method for outdoor large-scale scenes is realized based on this. Then, a complete contour estimation system is further designed based on the prediction results of the network. Given the shape characteristics of buildings of different sizes in the actual scene, all the identified building areas are divided into large-size buildings and small-size buildings according to the area. Finally, a set of post-processing and contour estimation algorithms for these two sizes of buildings are designed. In addition, a series of experiments are implemented on the test set of the public dataset to verify that the designed scheme not only has good structural features, but also has low algorithm complexity. It effectively avoids the limitations of traditional image feature matching methods. Moreover, the model is designed for mobile terminals, making full use of mobile computing resources, effectively reducing the dependence on the network, and avoiding network delays.

Although a suitable algorithm model is constructed, there are still many shortcomings. First, the detection of buildings mainly relies on visual algorithms, and it is difficult to achieve the best performance in scenes with severe visual conditions (such as evening, night, and other poor lighting conditions). Given the performance of the mobile terminal, the design size of the input image of the SqueezeNet SSD model is reduced to 224 * 224 * 3. When the distance of the building is too far, it may happen that the image is too small in the picture, resulting in missed detection. In the matching algorithm based on the idea of rotation angle, the VRA rotation angle in principle cannot represent the direction of the longitude and latitude coordinates of the center of the building, and it is just an approximation. Therefore, there must be a certain error between its numerical value and the GRA rotation angle, and the numerical value has a certain degree of fluctuation. Although this method can achieve accurate matching of building information in most cases, it may happen that the rotation angle matching fails in some complicated situations (such as multiple buildings being collinear or overlapping in shooting angles).

Data Availability

All data are fully available without restriction.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] H. Skinner, D. Sarpong, and G. R. White, "Meeting the needs of the Millennials and Generation Z: gamification in tourism through geocaching," *Journal of Tourism Futures*, vol. 4, no. 4, pp. 254–263, 2018.

[2] G. Li, D. Yanjie, Z. Hui, F. Xuebing, T. Xiangnong, and Z. Jinling, "Key technologies research and application of 3D modeling for digital city construction," *Bulletin of Surveying and Mapping*, vol. 96, no. 2, 2017.

[3] M. M. Rathore, A. Paul, W.-H. Hong, H. Seo, I. Awan, and S. Saeed, "Exploiting IoT and big data analytics: defining smart digital city using real-time urban data," *Sustainable Cities and Society*, vol. 40, pp. 600–610, 2018.

[4] Y. Liu, Z. Zhang, R. Zhong et al., "Multilevel building detection framework in remote sensing images based on convolutional neural networks," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 10, pp. 3688–3700, 2018.

[5] X. Pan and J. Zhao, "A central-point-enhanced convolutional neural network for high-resolution remote-sensing image classification," *International Journal of Remote Sensing*, vol. 38, no. 23, pp. 6554–6581, 2017.

[6] J. Yuan, Y. Wang, Y. Peng, and C. Wei, "Weak fault detection and health degradation monitoring using customized standard multiwavelets," *Mechanical Systems and Signal Processing*, vol. 94, pp. 384–399, 2017.

[7] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights Into Imaging*, vol. 9, no. 4, pp. 611–629, 2018.

[8] S. A. Mattonen, D. Gude, S. Echegaray, S. Bakr, D. L. Rubin, and S. Napel, "Quantitative imaging feature pipeline: a web-based tool for utilizing, sharing, and building image-processing pipelines," *Journal of Medical Imaging (Bellingham, Wash.)*, vol. 7, no. 4, pp. 42803–42811, 2020.

[9] E. Bottani and G. Vignali, "Augmented reality technology in the manufacturing industry: a review of the last decade," *IISE Transactions*, vol. 51, no. 3, pp. 284–310, 2019.

[10] M. B. Ibáñez, A. Uriarte Portillo, R. Zatarain Cabada, and M. L. Barrón, "Impact of augmented reality technology on academic achievement and motivation of students from public and private Mexican schools. A case study in a middle-school geometry course," *Computers & Education*, vol. 145, pp. 103734–103742, 2020.

[11] X. Xiao, L. Jin, Y. Yang, W. Yang, J. Sun, and T. Chang, "Building fast and compact convolutional neural networks for offline handwritten Chinese character recognition," *Pattern Recognition*, vol. 72, pp. 72–81, 2017.

[12] X. Yan, T. Ai, M. Yang, and H. Yin, "A graph convolutional neural network for classification of building patterns using spatial vector data," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 150, pp. 259–273, 2019.

[13] J. Wang, J. Qin, J. Qin, X. Xiang, Y. Tan, and N. Pan, "CAPTCHA recognition based on deep convolutional neural network," *Mathematical Biosciences and Engineering*, vol. 16, no. 5, pp. 5851–5861, 2019.

[14] W. Wei, Y. Wong, Y. Du, Y. Hu, M. Kankanhalli, and W. Geng, "A multi-stream convolutional neural network for sEMG-based gesture recognition in muscle-computer interface," *Pattern Recognition Letters*, vol. 119, pp. 131–138, 2019.

[15] G. Yao, T. Lei, and J. Zhong, "A review of convolutional-neural-network-based action recognition," *Pattern Recognition Letters*, vol. 118, pp. 14–22, 2019.

[16] Y. Zhou, H. Luo, and Y. Yang, "Implementation of augmented reality for segment displacement inspection during tunneling construction," *Automation in Construction*, vol. 82, pp. 112–121, 2017.

[17] G. Mylonas, C. Triantafyllis, and D. Amaxilatis, "An augmented reality prototype for supporting IoT-based educational activities for energy-efficient school buildings," *Electronic Notes in Theoretical Computer Science*, vol. 343, pp. 89–101, 2019.

[18] Y.-J. Chen, Y.-S. Lai, and Y.-H. Lin, "BIM-based augmented reality inspection and maintenance of fire safety equipment," *Automation in Construction*, vol. 110, pp. 103041–103051, 2020.

[19] I. García-Pereira, C. Portalés, J. Gimeno, and S. Casas, "A collaborative augmented reality annotation tool for the inspection of prefabricated buildings," *Multimedia Tools and Applications*, vol. 79, no. 9, pp. 6483–6501, 2020.

[20] D. Liu, X. Xia, J. Chen, and S. Li, "Integrating building information model and augmented reality for drone-based building inspection," *Journal of Computing in Civil Engineering*, vol. 35, no. 2, pp. 4020073–4020084, 2021.

[21] M. Zhang, W. Li, Q. Du, L. Gao, and B. Zhang, "Feature extraction for classification of hyperspectral and LiDAR data using patch-to-patch CNN," *IEEE Transactions on Cybernetics*, vol. 50, no. 1, pp. 100–111, 2018.

[22] L. You, Q. Peng, Z. Xiong, D. He, M. Qiu, and X. Zhang, "Integrating aspect analysis and local outlier factor for intelligent review spam detection," *Future Generation Computer Systems*, vol. 102, pp. 163–172, 2020.

[23] S. K. Erskine and K. M. Elleithy, "Real-time detection of DoS attacks in IEEE 802.11p using fog computing for a secure intelligent vehicular network," *Electronics*, vol. 8, no. 7, pp. 776–784, 2019.

[24] H. Ibrahem, A. D. A. Salem, and H.-S. Kang, "Real-time weakly supervised object detection using center-of-features localization," *IEEE Access*, vol. 9, pp. 38742–38756, 2021.

[25] F. Hou, W. Lei, S. Li, J. Xi, M. Xu, and J. Luo, "Improved Mask R-CNN with distance guided intersection over union for GPR signature detection and segmentation," *Automation in Construction*, vol. 121, pp. 103414–103426, 2021.

[26] L. Wang, Y. Xu, Y. Li, and Y. Zhao, "Voxel segmentation-based 3D building detection algorithm for airborne LIDAR data," *PLoS One*, vol. 13, no. 12, pp. e0208996–e0209006, 2018.

[27] S. Dwivedi, M. Vardhan, and S. Tripathi, "Building an efficient intrusion detection system using grasshopper optimization algorithm for anomaly detection," *Cluster Computing*, vol. 277, pp. 1–20, 2021.

[28] J. Wu and J. Xu, "Research on noise impact of building environment based on FPGA high-performance algorithm," *Microprocessors and Microsystems*, vol. 80, pp. 103342–103354, 2021.